REUSE READINESS LEVELS AS A MEASURE OF SOFTWARE REUSABILITY

James J. Marshall[1] *and Robert R. Downs*[2]

[1]Innovim/NASA Goddard Space Flight Center, [2]Columbia University

ABSTRACT

The reuse of software and related artifacts offers the potential for cost savings in various industries and has contributed to the development of the cyberinfrastructure that is used by the Earth science community. Developing measures that enable the assessment of software in terms of its potential reusability can contribute to efforts of both developers and reusers of software. Draft Reuse Readiness Levels (RRLs) have been developed as an instrument for assessing the maturity of software products for potential reuse. The process employed to develop the draft RRLs is described, and the initial summary of topic areas defined for each level is presented. Based on recommendations from the community of Earth science data systems software developers, areas for improving the precision of the draft RRLs have been identified and an approach for identifying enhancements is described.

Index Terms**—**Software reuse, reuse readiness level, Earth science, cyberinfrastructure, eScience, NASA

1. INTRODUCTION

The value of software reuse can be readily observed by the growth of the open source community of developers and reusers of software, which has been changing practices for software development in various industries [1]. Likewise, the community of Earth science systems developers has benefited from the development and reuse of reusable software [2]. Furthermore, eScience and the Earth science community in particular also have benefited from the adoption of open source software and standards [3].

The widespread reuse of existing software needs to be supported by tools that can assist both developers who create reusable software and users who reuse existing software. Software developers need support to guide their preparation of software for potential reuse. Similarly, prospective users of software need support when assessing software for potential reuse. Both software developers and software adopters share a common need for capabilities to evaluate software and related assets for potential reuse. Measures are needed to meet the common needs of both developers and reusers to assess software assets for potential reusability. The identification and validation of such measures requires research [4].

Providing a common instrument for developers and users of software to measure the maturity of software for reuse can provide both groups with uniform support that meets the needs of both groups. Common measures are needed so that each group can assess software, at any point during its development, to determine the extent to which the software is ready for reuse.

The reusability of software products is a factor often neglected in common measures of technology maturity such as Technology Readiness Levels (TRLs), for which the United States Department of Defense (DoD) and the National Aeronautics and Space Administration (NASA) have two of the most often used definitions [5, 6]. Measures such as TRLs are concerned primarily with usability of a technology, not its reusability. TRLs have been applied to the adoption of new technologies within various domains and have inspired the development of tools to automate the assessment of technology and reduce the costs of evaluating candidate technologies that are being considered for adoption [7].

Reusability is a separate issue and an important one to consider, since the reuse of software can provide many benefits such as saving time, saving money, and increasing reliability [8]. Software developers would benefit greatly from having a simple way to quickly estimate the readiness of software assets to be reused. While an estimation of reuse readiness will not eliminate the need for examination and testing of candidate reusable assets, it will enable developers to more easily determine how ready the software is for their purposes and how much modification may be necessary before it can fill their needs.

In addition to the TRLs, various models for assessing the maturity of software have been proposed, including the Business Readiness Rating [9], the Open BQR [10], and the Open Source Maturity Model [11]. However, in the absence of widespread adoption of any of these models by the large community that reuses software, an alternative model is being developed to assess the maturity of software for reuse.

The NASA Earth Science Data Systems (ESDS) Software Reuse Working Group is in the process of developing a set of Reuse Readiness Levels (RRLs) for the purpose of determining the reuse maturity of software assets. These levels are modeled after NASA's TRLs, which have been used for many years, particularly for assessing hardware readiness for space flight purposes. The availability of RRLs will enable software developers to evaluate software that is being prepared for potential reuse and serve as a resource for prospective users to select software to be reused.

This paper describes the background and justification for the creation of RRLs and work done to date on the development of both the topic levels and the single RRL scale. Community feedback is considered, its relation to the measurement of reuse maturity interpreted, and some improvements for the next revision of the RRL scale are suggested. Some of the potential uses, applications, and implications of RRLs are also described.

2. REUSE READINESS LEVEL TOPIC AREAS

An iterative approach is being taken to develop the RRLs. Initially, the needs of both software providers and software reusers were identified to determine the capabilities to be offered. Software providers need to determine whether their software can be used by others, whether the software is ready for reuse, and which areas need to be improved for use by others. Software reusers need to determine whether to consider reusing a software asset, compare software assets available for reuse, assess strengths and weaknesses of such software, and identify where additional development is necessary for reuse.

Software development characteristics determined to be critical for enabling software reuse were identified and categorized as initial topics to be considered for the development of the RRLs. Different types of software programs were considered during the analysis. Software development characteristics, identified as both critical and challenging for software reuse, were nominated as the initial candidate topics for the RRLs. These included Documentation, Extensibility, Intellectual Property Issues (including licensing), Modularity, Packaging, Portability (including generality), Standards Compliance, Support, and Verification/Testing.

For the preparation of the topic areas, teams of experts were organized to analyze each of the candidate topic areas identified for the RRLs. Each team, consisting of at least two individuals who have been involved in software development, prepared the topic area in which they possessed expertise for inclusion in the RRLs.

Initially, a description was prepared for each of the nominated topics. The descriptions defined each topic area and described aspects of software development that differentiated it from other topics. Each initial RRL topic area was analyzed to identify and describe break points or intermediate work products completed during software development that could contribute to the potential for software reuse. The descriptions of each successive work product include aspects of the sequential process, the resulting work product, and implications for reuse of the work product, the topic area, and the software. In addition, a label was created for each sequential work product to succinctly describe the work product and its potential for reuse.

Next, each work product that had been identified as contributing to software reuse was assigned a sequential number to reflect the sequence of activities that occurred during the software development process. Consecutively increasing numbers assigned to each successive work product created during software development reflected the increasing maturity that the software attained during preparation for reuse. Five to nine successive work products were identified for each of the nine topics previously identified and these were nominated as potential levels in the sequence that occurs during the software development process for each respective topic area.

3. REUSE READINESS LEVEL SUMMARIES

All of the potential levels and each of the topic areas were combined in a nine-by-nine matrix, where each column represented a topic area and each row represented a tentative level in the sequential development of the software. For topic areas that contained nine levels, each sequential level was assigned to a consecutive row in the matrix. Topic areas that contained less than nine levels were analyzed to determine corresponding levels that described similar aspects of maturity completed for the other topic areas. Where some topic area levels did not correspond directly with other levels, additional analysis identified the appropriate row for assigning the potential level for the topic. Combining the categories, levels, and descriptions resulted in a single proposed Reuse Readiness Level (RRL) scale.

When all of the potential levels for each of the topic areas had been assigned to rows within the matrix, four experts in software development each conducted an independent review and analysis of each level. The analysis of each level reviewed each topic area represented within the level and was conducted to determine whether the activities described within each level were consistent. Each of the reviewers reported on the inconsistencies observed within each level and prepared a draft statement that summarized the software development achieved for each level along with the limitations and the potential for reuse of the software that was observed for each level. Review of each of the topic area categories across each level resulted in a draft summary describing each proposed level as software becomes more mature during preparation for potential reuse.

A fifth reviewer, also an expert in software development, prepared a single statement that summarized each level, based on the four independently produced summary statements. The fifth reviewer analyzed each of the draft statements summarizing each level and identified common terminology among each pair of summaries. Terms found to be consistent with those used in the descriptions of the corresponding levels were included in the summaries prepared for each level. The current draft of these RRL level summaries is shown in Table 1 below.

Table 1 – Summary of Draft Reuse Readiness Levels

RRL	Summary
1	No reusability; the software is not reusable.
2	Initial reusability; software reuse is not practical.
3	Basic reusability; the software might be reusable by skilled users at substantial effort, cost, and risk.
4	Reuse is possible; the software might be reused by most users with some effort, cost, and risk.
5	Reuse is practical; the software could be reused by most users with reasonable cost and risk.
6	Software is reusable; the software can be reused by most users although there may be some cost and risk.
7	Software is highly reusable; the software can be reused by most users with minimum cost and risk.
8	Demonstrated reusability; the software has been reused by multiple users.
9	Proven reusability; the software is being reused by many classes of users over a wide range of systems.

4. IMPROVING THE REUSE READINESS LEVELS

Presentations on drafts of the RRLs [12, 13, 14] resulted in suggestions for the draft topic levels and the initial RRL summary levels. These suggestions also highlighted a number of issues that needed to be addressed to improve the RRLs for use by developers and users as the RRL scale continues to be revised and developed. These include issues related to security, cost, risk, and lifecycle issues as well as ensuring the levels have quantitative measures for consistent and repeatable assessment of maturity. In addition, the RRLs must be sure to cover all types of reuse (e.g., white box and black box [2]), focus on reusability while still including some degree of usability, and be designed so that different audiences can make use of them (e.g., software developers and project managers may both want a measure of software reusability, but have different needs/desires for this measure). Based on the recommendations received, an approach was identified for improving the draft RRLs to address these issues.

Initially, all of the comments and suggestions for improvement, which were received from the community for each topic area and for the RRLs in general, were presented to potential reviewers and discussed to inform the next round of reviews. Next, individual reviewers each analyzed one or more topic areas and provided recommendations to differentiate the changes in software that must be completed to attain each successive level for facilitating reuse. All topic areas were covered by these individual reviews. Differentiating changes between levels required the identification of terminology that can describe software qualifications clearly to enable the assignment of a level when evaluating the reusability of software for a topic area.

The recommendations for each RRL topic area were compiled into a single document to facilitate a complete and consistent analysis of all the RRLs. Recommendations pertaining to more than one topic area were included in each area to which they applied. The comprehensive RRL document was then analyzed by three reviewers who suggested additional changes for each topic area, as necessary, and for each Summary Description of the RRLs, to address the recommendations of the individual reviewers. They also accepted recommendations from the individual reviewers that sufficiently addressed previously received feedback on the topic areas. Also, each level was analyzed for inconsistencies between topic areas within each level. While a software product could be immature for reuse in one topic area while achieving maturity in another, the relationships among the topic areas require refinement at each level to ensure that contradictions of reuse readiness do not exist within any particular level.

The revised draft RRLs will be presented to the community of Earth science data systems developers to identify additional improvements that are needed. This process of revising the RRLs in response to community feedback will result in a community-developed measurement of the reusability of software that can offer a number of potential uses for reuse efforts (see Section 5).

In addition, an RRL calculator is being developed concurrently to provide a Web-based tool for developers and potential users of software to determine the reusability of a particular software product. The development of the RRL calculator is expected to enable testing of subsequent drafts of the RRLs and foster the identification of topic areas within each level that require further specification to enable precise measurement and ease of use.

5. SUMMARY

Software reuse contributes to the development of cyberinfrastructure to support eScience and offers potential benefits for the Earth science data systems development community. The RRLs are being developed to offer capabilities for developers and reusers of software to measure the reusability of software in terms of several dimensions, which have been identified by members of the Earth science data systems development community as characteristics that are needed to support software reuse. The RRLs are presented as a matrix of topic areas and levels that can be used to determine the reusability of software being considered for reuse.

The topic areas of the RRLs are being reviewed to identify improvements for reducing ambiguity and to specify measurable characteristics within each level for assessing the reusability of software. Each level of the RRLs is being reviewed for consistency among the topic areas.

Potential uses of the RRLs are as: measures for assessing and comparing software assets for potential reuse and for determining potential costs of their reuse; metadata for reusable software assets stored in catalogs and repositories, as a guide to reusers; an indicator of areas to focus on when creating reusable assets, as a guide to providers; and part of requests for proposals or contracts, asking for a reuse approach or how assets are being made reusable [14]. When the RRLs are fully developed, they will assist prospective reusers in selecting appropriate software to reuse, as well as assisting developers in ensuring that their assets are reusable by others. While initially developed for use in the domain of Earth science, the RRLs are general enough to be used in other domains, providing the potential for use in any software development project.

6. ACKNOWLEDGEMENTS

The authors appreciate the contributions of the current and previous members of the Earth Science Data Systems (ESDS) Software Reuse Working Group on the development of the draft RRLs, and especially appreciate the thoughtful comments for improving earlier drafts of the paper recommended by Neil Gerard and Ryan Gerard of Innovim/NASA.

The authors very much appreciate the support received from the National Aeronautics and Space Administration (NASA) for the work reported in this paper, including the support for Robert Downs, under Contract NAS5-03117.

7. REFERENCES

[1] D. Riehle, "The Economic Motivation of Open Source Software: Stakeholder Perspectives", Computer, 40(4), IEEE, http://hdl.handle.net/10.1109/MC.2007.147, pp. 25-32, April 23, 2007.

[2] S. Samadi, N. Almaeh, R. Wolfe, S. Olding, and D. Isaac, 'Strategies for enabling software reuse within the Earth Science Community", Proceedings of the 2004 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), IEEE, http://hdl.handle.net/10.1109/IGARSS.2004.1370796, pp. 2196-2199, December 27, 2004.

[3] M.K. Ramamurthy, "A new generation of cyberinfrastructure and data services for earth science education and research. Advances in Geosciences, 8, Copernicus GmbH, http://www.adv-geosci.net/8/69/2006/adgeo-8-69-2006.pdf, pp. 69-78, June 6, 2006.

[4] W. B. Frakes and K. Kang. "Software Reuse Research: Status and Future", 31(7), Transactions on Software Engineering, IEEE, http://doi.ieeecomputersociety.org/10.1109/TSE.2005.85, pp. 529-536, July 2005.

[5] Deputy Under Secretary of Defense for Science and Technology, "Technology Readiness Assessment (TRA) Deskbook," http://www.dod.mil/ddre/doc/tra_deskbook_2005.pdf, May 2005.

[6] J.C. Mankins, "Technology Readiness Levels: A White Paper," Advanced Concepts Office, Office of Space Access and Technology, National Aeronautics and Space Administration, (NASA). http://www.hq.nasa.gov/office/codeq/trl/trl.pdf, April 6, 1995.

[7] B.L. Britt, M.W. Berry, M. Browne, M.A. Merrell, and J. Kolpack, "Document Classification Techniques for Automated Technology Readiness Level Analysis", Journal of the American Society for Information Science and Technology, 59(4), Wiley Interscience, http://hdl.handle.net/10.1002/asi.20770, pp. 675-680, January 4, 2008.

[8] J.J. Marshall, S.W. Olding, R.E. Wolfe, and V.E. Delnore, "Software Reuse Within the Earth Science Community," *Geoscience and Remote Sensing Symposium, 2006. IGARSS 2006. IEEE International Conference on,* pp. 2880–2883, July 31, 2006–Aug. 4, 2006.

[9] OpenBRR.org, "Business Readiness Rating for Open Source", http://www.openbrr.org/wiki/images/d/da/BRR_whitepaper_2005RFC1.pdf, 2005.

[10] D. Taibi, L. Lavazza, and S. Morasca, "OpenBQR: a framework for the assessment of OSS," *IFIP International Federation for Information Processing*, Springer, Boston, pp. 173–186, 2007.

[11] Navica, "Open Source Maturity Model", http://www.navicasoft.com/pages/osmm.htm.

[12] Software Reuse Working Group, "Reuse Readiness Levels (RRLs) – A Work in Progress," presentation at the 6th ESDS WG Meeting, October 2007. http://www.esdswg.com/softwarereuse/Resources/library/6th-esds-wg-meeting/ESDSWG6RRLSlides.pdf/view

[13] J.J. Marshall, S.W. Berrick, A. Bertolli, H. Burrows, V.E. Delnore, R.R. Downs, Y. Enloe, S. Falke, M. Folk, N. Gerard, R. Gerard, M. Hunter, T. Jasmin, D. McComas, S. Samadi, M. Sherman, R. Swick, C. Tilmes, and R.E. Wolfe, "Reuse Readiness Levels (RRLs): Proposed Topic Area Levels," poster presented at the 6th ESDS WG Meeting, October 2007.

[14] J.J. Marshall, S.W. Berrick, A. Bertolli, H. Burrows, V.E. Delnore, R.R. Downs, Y. Enloe, S. Falke, M. Folk, N. Gerard, R. Gerard, M. Hunter, T. Jasmin, D. McComas, S. Samadi, M. Sherman, R. Swick, C. Tilmes, and R.E. Wolfe (2007), "A Community-Developed Measurement of the Reusability of Software Through Reuse Readiness Levels", *Eos Trans. AGU, 88*(52), Fall Meet. Suppl., Abstract IN31A-0074.